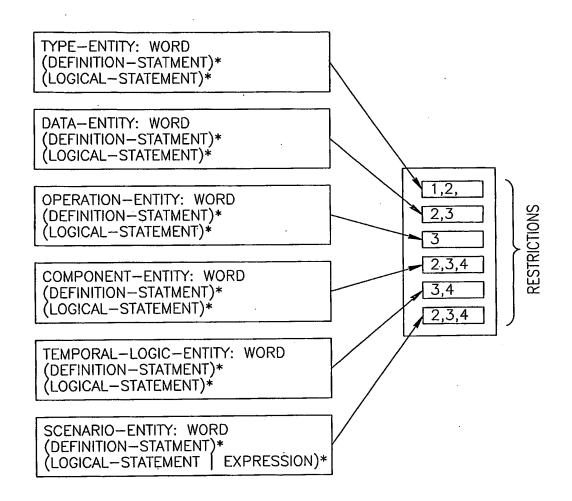


SUBSTITUTE SHEET (RULE 26)



RESTRICTIONS

- 1. DEFINITION-STATEMENT MUST BE (*:=* TYPE-STATEMENT)
- 2. NO INPUT/OUTPUT DEFINITION STATEMENT DEVARIATION ALLOWED
 3. DEFINITION—STATEMENT MUST NOT BE (*:=* TYPE—STATEMENT)
 4. TWO DEFINITION—STATEMENTS SECTIONS

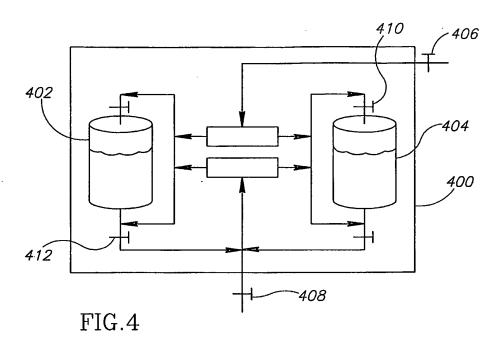
FIG.2

-302 EXPRESSION:== VALUE (OP-SYMBOL EXPRESSION)? | TYPE-STATEMENT TYPE-STATMENT:==TYPE-NAME | TYPE-STATEMENT (cartesian|function)TYPE-STATEMENT DEFINITION-STATMENT:==(INPUT | OUTPUT)? DEF-NAME DEF-SYMBOL DEF-EXPRESSION OP-SYMBOL:==arithmatic operations set operations, sequence operations. (REL|QUAN-PREDICATE | true | false) QUAN—PREDICATE:==quantifier DEFINITION—STATEMENT PREDICATE DEF-EXPRESSION:=PREDICATE | EXPRESSION | BOOLEAN-EVENT VALUE:==NUM, WORD, TUPLE, SET, VARIABLE, FUNCTION-CALL REL:==EXPRESSION relation—symbol EXPRESSION implies | iff _OGICAL-STATEMENT:==PREDICATE STAT-SYMBOL PREDICATE ō PREDICATE LOĞICAL—CONNECTÍVÉ TYPE-STATMENT:==TYPE-NAME | (powerset|seq) TYPE-STATEMENT |YPE-NAME:==WORD PREDICATE:==(not PREDICATE) OGICAL—CONNECTIVE:==and | OP-SCHEMA-CALL DEF-NAME:==WORD DEF-SYMBOL:==":", OUTPUT:=="output" STAT-SYMBOL:=='INPUT:=="input"

FIG.3

-306 -305 BOOL-EVENT:==not BOOL-EVENT | BOOL-EVENT((and | or | implies | iff)BOOL-EVENT)? OPERATION-SEQ *arithmatic operators, set operation, sequence operations — mathemathical operators <u>_</u> OPERATION-SEQ *cartesian, function powerset, seq-mathemathical type symbols |TEMPORAL-STATEMENT:==(not)? OPERATION-SEQ | OPERATIC | OPERATION-SEQ:==BOOL-EVENT |("*") (SEP BOOL-EVENT)* |SEP:=="," "<<"(WORD)+">>" *relation—symbol mathema—tical relation symbols *quantifier— "forall", "exists" $\mathsf{WORD}:==("a"-"z")*\mathsf{TUPLE}:=="("VALUE(","VALUE)*)$ and, or, not implies, iff - logical operators WORD OP-SCHEMA-CALL:=="call" FUNC-CAL FUNC-CALL:=WORD"("(VARIABLE)*")" BOOLEAN-EVENT:==PREDICATE *true/false — boolean values BOOL-EVENT (NUM) $SET:=="\{"(VALUE)^*"\}$ variablè:≟=word *(6-0)==:MNN

FIG.3



```
COMPONENT ENTITY:

StorageContainer

VARIABLES:

container: CONTAINER;

INITIALIZATION:

st1, st2: STORAGETANK;

st1=STORAGETANK (100);
st2=STORAGETANK (100);
container = CONTAINER(st1,st2);

EXPLANATION:
```

FIG.5

DATA ENTITY:

Container

VARIABLES:

st1, st2: STORAGETANK;

INVARIANTS:

EXPLANATION:

FIG.6

DATA ENTITY:

StorageTank

VARIABLES:

contents : naturals; capacity : naturals;

INVARIANTS:

capacity >= contents

EXPLANATION:

FIG.7

TYPE ENTITY:

SpecificationTypes

TYPE DEFINITIONS:

naturals : integers;

IndicationMessage:= {SUCCESS, FAIL}

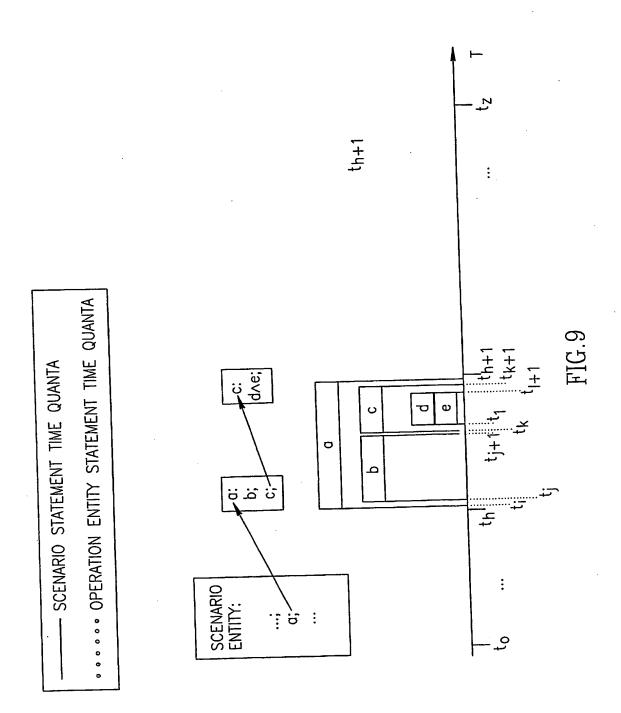
RESTRICTIONS:

 \forall n:naturals • n>=0;

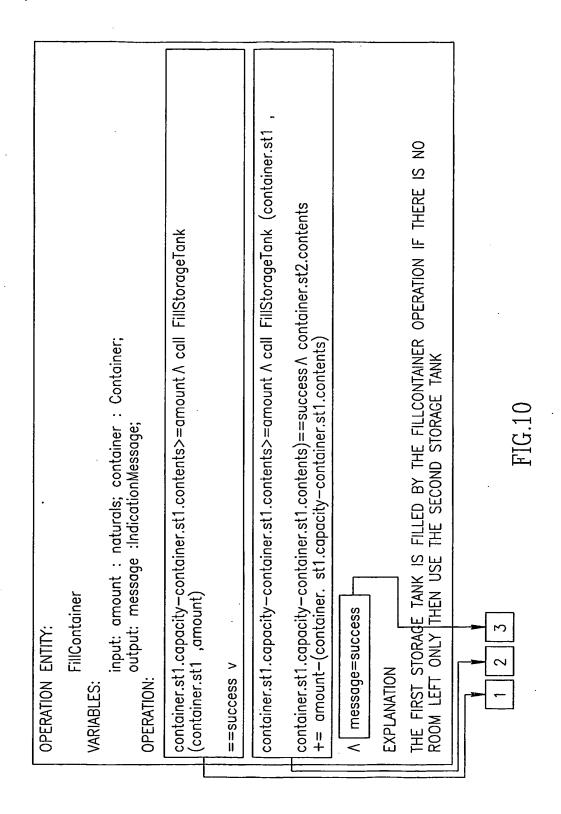
EXPLANATION:

FIG.8

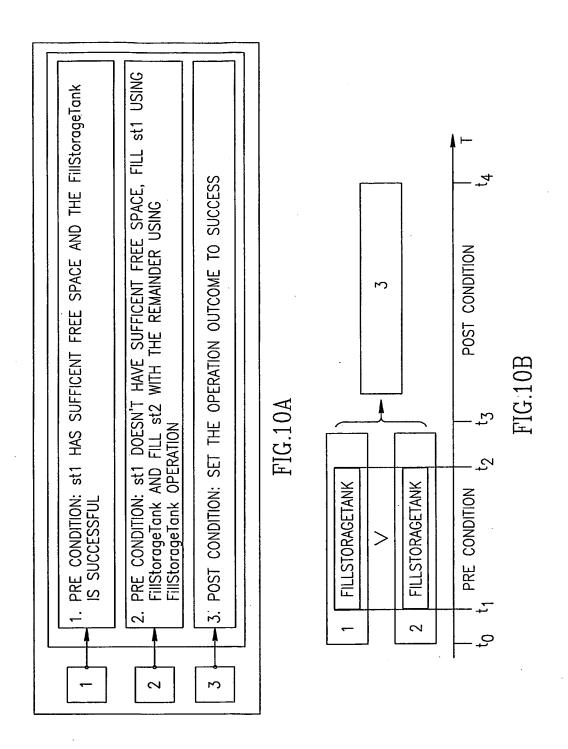
7/29



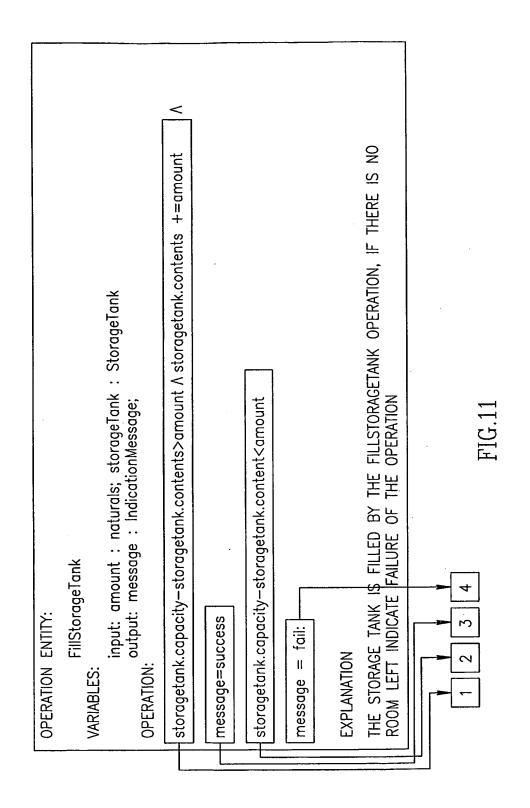
SUBSTITUTE SHEET (RULE 26)



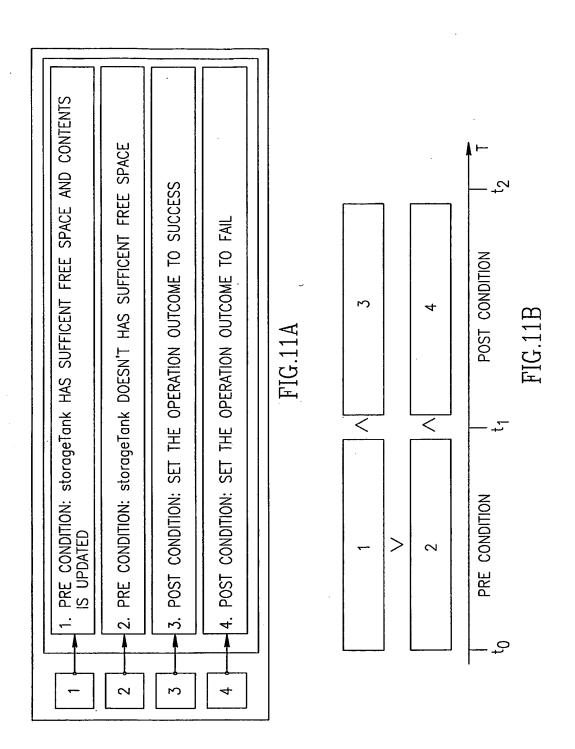
SUBSTITUTE SHEET (RULE 26)



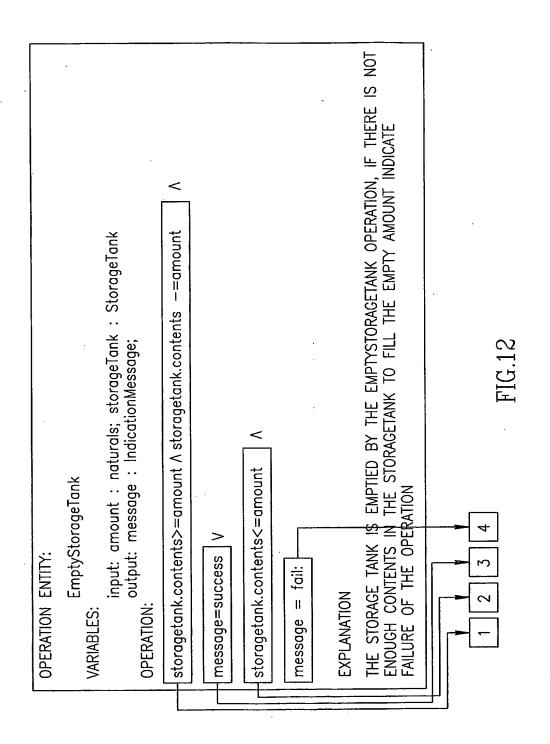
SUBSTITUTE SHEET (RULE 26)



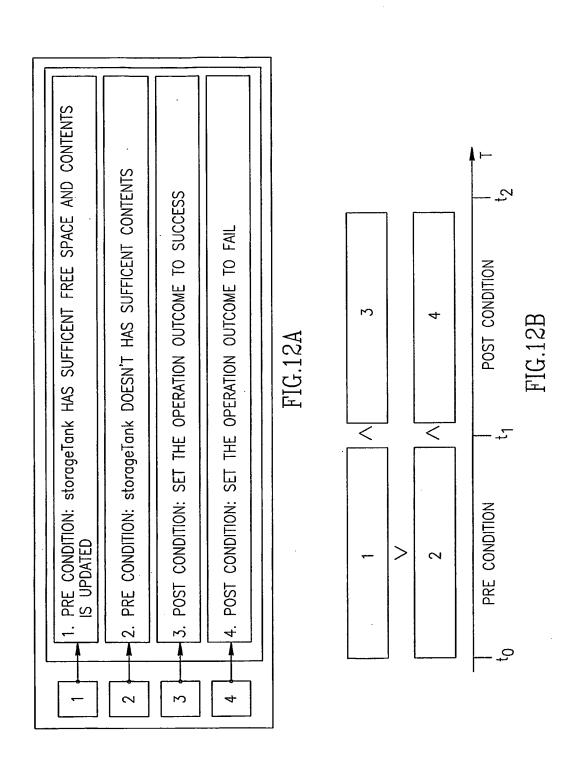
SUBSTITUTE SHEET (RULE 26)



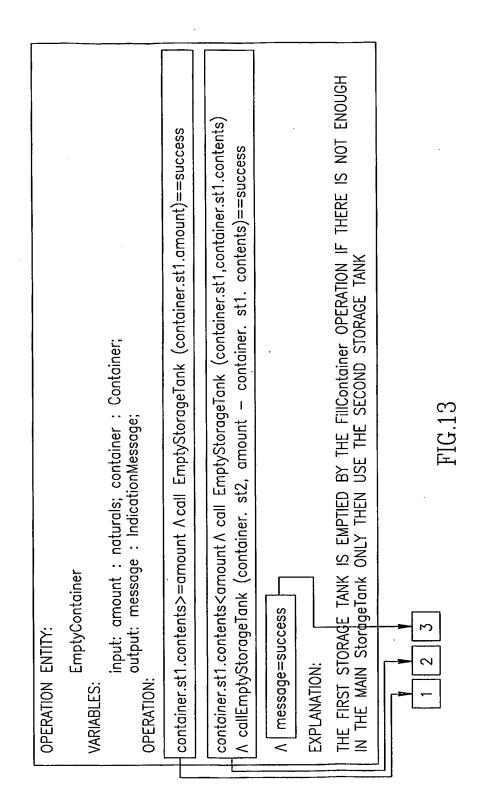
SUBSTITUTE SHEET (RULE 26)



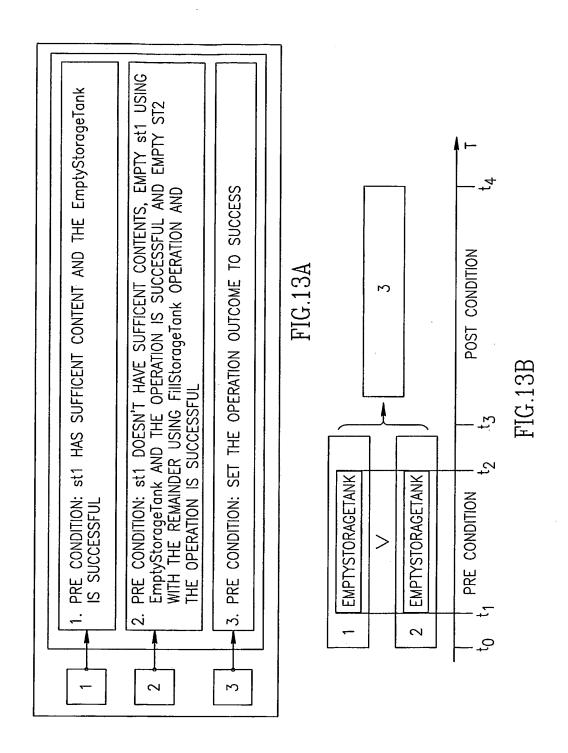
SUBSTITUTE SHEET (RULE 26)



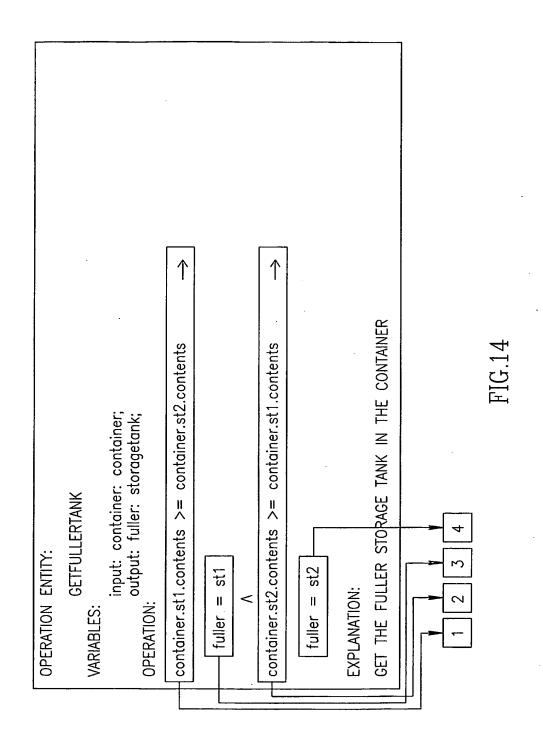
SUBSTITUTE SHEET (RULE 26)



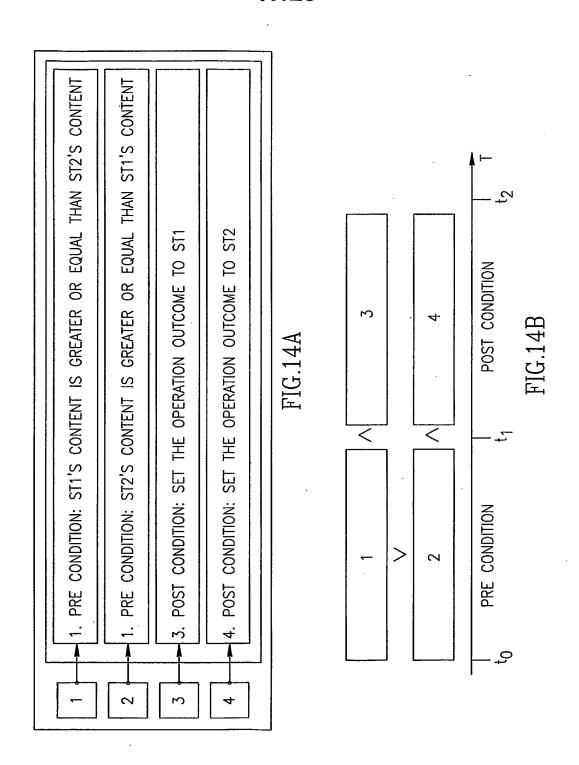
SUBSTITUTE SHEET (RULE 26)



SUBSTITUTE SHEET (RULE 26)

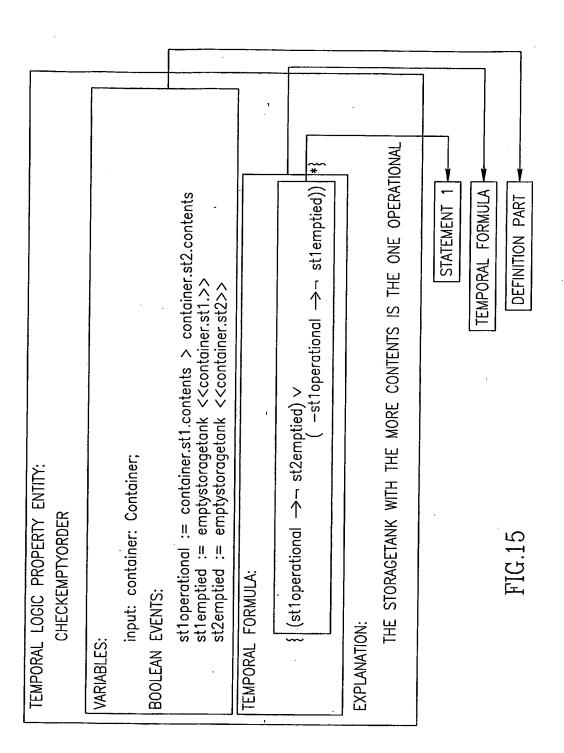


SUBSTITUTE SHEET (RULE 26)

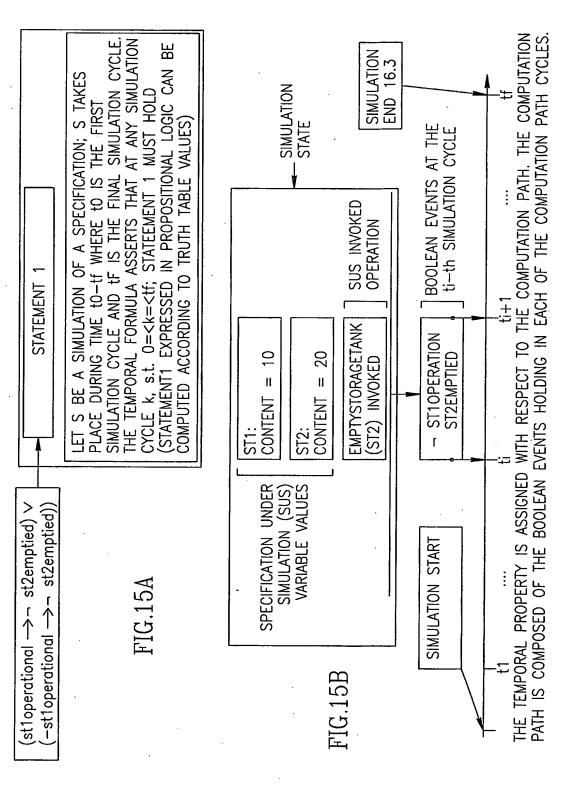


SUBSTITUTE SHEET (RULE 26)

18/29



SUBSTITUTE SHEET (RULE 26)



SUBSTITUTE SHEET (RULE 26)

```
test1

COMPONENT:

StorageContainer;

EXTERNAL VARIABLES:

SCENARIO:

call fillcontainer(storagecontainer.container, 100);

call fillcontainer(storagecontainer.container, 100);

call getfullertank();
```

FIG.16

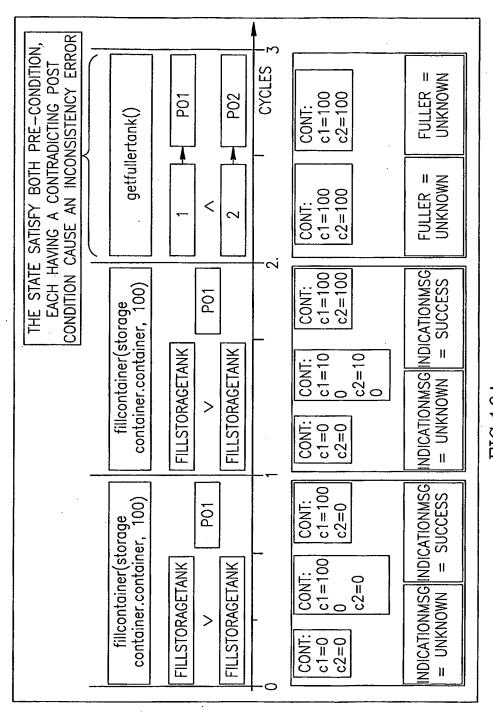


FIG.16A

SUBSTITUTE SHEET (RULE 26)

```
SCENARIO EXECUTION ENTITY:

test2

COMPONENT:

storagecontainer;

EXTERNAL VARIABLES:

SCENARIO:

call fillcontainer(storagecontainer.container, 50);

call fillcontainer(storagecontainer.container, 50);

call fillcontainer(storagecontainer.container, 100);

call fillcontainer(storagecontainer.container, 100);
```

FIG.17

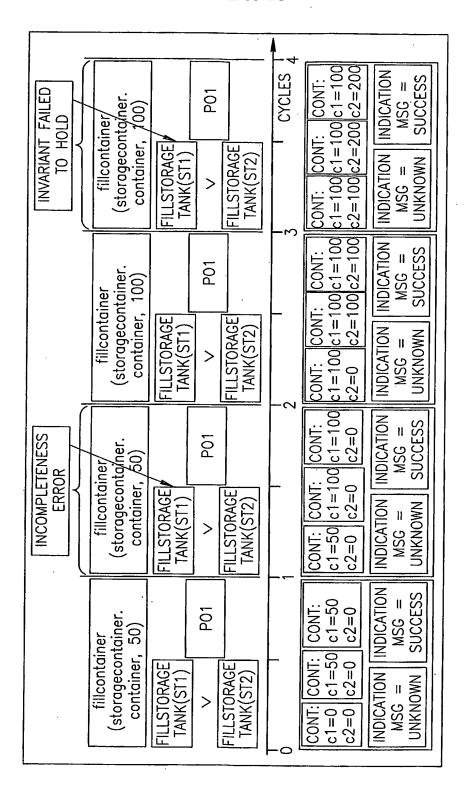


FIG.17A

SUBSTITUTE SHEET (RULE 26)

SCENARIO EXECUTION ENTITY:
test3
COMPONENT:
storagecontainer;
EXTERNAL VARIABLES:
chkr : checkemptyorder
SCENARIO:
call fillcontainer(storagecontainer.container, 200) ;
call emptycontainer(storagecontainer.container, 90) ;
call emptycontainer(storagecontainer.container, 10) ;
call emptycontainer(storagecontainer.container, 101);
TEMPORAL LOGIC PROPERTY INSTANCE

FIG.18

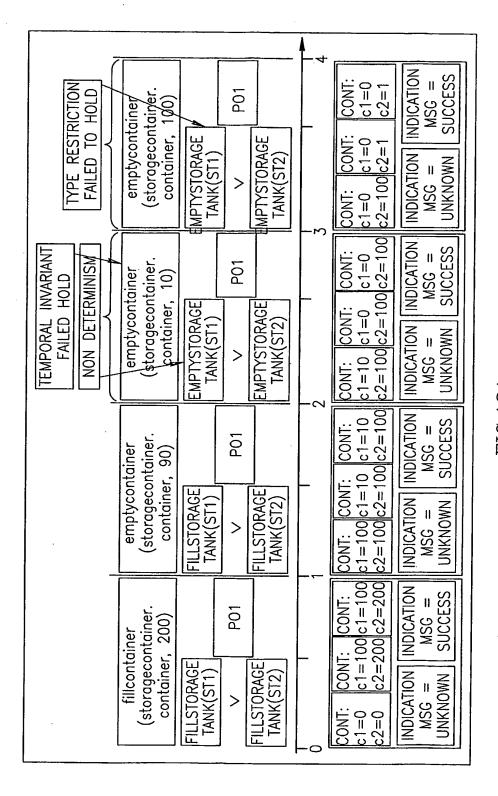
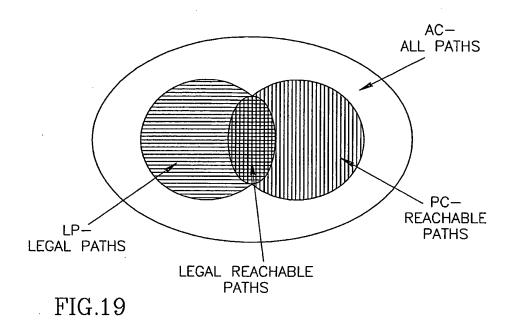


FIG.18A

WO 2005/043278 PCT/IL2004/000990

26/29



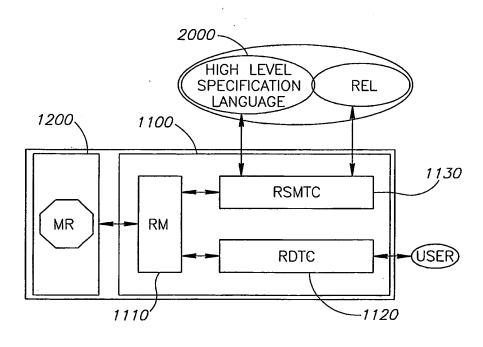
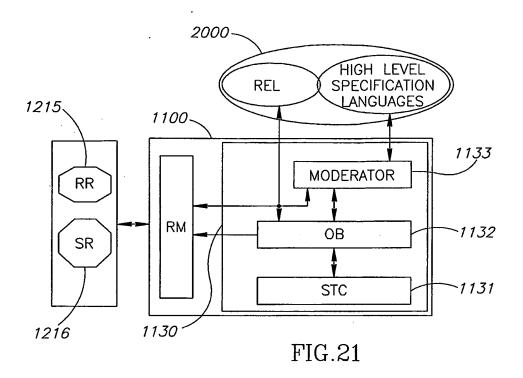


FIG.20

WO 2005/043278 PCT/IL2004/000990

27/29



1121.a 1121.b 1121.c SR SVM DVM 1121.c RQTM 1122

SUBSTITUTE SHEET (RULE 26)

FIG.22

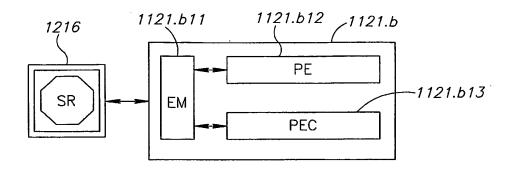


FIG.23

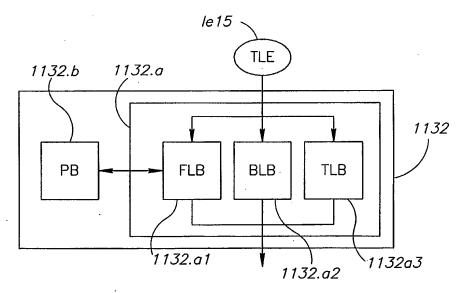


FIG.24

WO 2005/043278 PCT/IL2004/000990

29/29

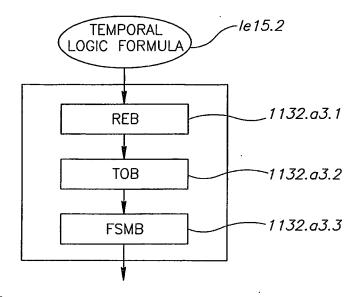


FIG.25

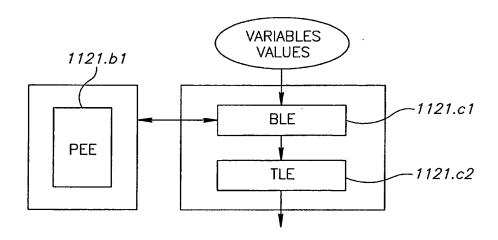


FIG.26